

(19)日本国特許庁 (JP)

(12)特許公報 (B2)

(11)特許番号

第2850707号

(45)発行日 平成11年(1999)1月27日

(24)登録日 平成10年(1998)11月13日

(51)Int.Cl.⁶
G 1 0 H 1/00
G 0 6 F 17/10
G 1 0 H 1/02

識別記号

F I
G 1 0 H 1/00
1/02
G 0 6 F 15/31

Z
D

請求項の数1(全10頁)

(21)出願番号 特願平5-143160
(22)出願日 平成5年(1993)6月15日
(65)公開番号 特開平7-13561
(43)公開日 平成7年(1995)1月17日
審査請求日 平成8年(1996)6月27日

(73)特許権者 000004075
ヤマハ株式会社
静岡県浜松市中沢町10番1号
(72)発明者 東 岩男
静岡県浜松市中沢町10番1号 ヤマハ株
式会社内
(74)代理人 介理士 高橋 敬四郎 (外1名)

審査官 渡邊 聰

(56)参考文献 特開 平2-257198 (JP, A)
特開 昭60-39237 (JP, A)
特開 平4-219045 (JP, A)

(58)調査した分野(Int.Cl.⁶, DB名)
G10H 1/02

(54)【発明の名称】 楽音制御装置

1

(57)【特許請求の範囲】

【請求項1】 音源回路から供給される楽音信号を制御するための楽音制御装置であって、外部に記憶されたプログラムに従って演算処理を行なうCPUと、内部に記憶したマイクロプログラムに従って演算処理を行なうDSPと、前記CPUおよびDSPからアクセス可能なメモリと、前記CPUからアクセス可能な他の回路と、前記CPUから前記メモリへのアクセスと前記DSPから前記メモリへのアクセスとが同時に発生した時には前記CPUにウェイト信号を供給して前記DSPから前記メモリへのアクセスを優先させ、前記CPUから前記他の回路へのアクセスと前記DSPから前記メモリへのアクセスとが同時に発生した時には、前記CPUにウェイ

2

ト信号を供給しないアクセス制御手段とを有する楽音制御装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、メモリを利用する電子回路に関し、特にCPU(中央演算処理装置)とDSP(デジタルシグナルプロセッサ)とを用いて楽音信号を制御する楽音制御装置に関する。

【0002】

【従来の技術】 電子楽器においては、楽音信号を発生、制御するため、CPUが広く利用されている。処理プログラムをROM(読み専用メモリ)に記憶させ、RAM(ランダムアクセスメモリ)をレジスタ類等として用い、CPUでプログラムを実行して、音源回路に楽音信号を発生させる。

【0003】近年、発生する楽音に対する高度化、多様化の要求に対応して信号処理量が増大し、信号処理の高速化が要求されている。これらの要求に応えるため、特にリバーブ/残響等の効果付与の用途にDSPが用いられるようになってきた。

【0004】図7に、従来技術による電子楽器の構成例を示す。図において、CPUバス51にCPU53、ROM、RAM等のメモリ61、音源回路54と共にDSP55が接続されている。また、1/F64を介して鍵盤65、音色切り替えスイッチ67等も接続されている。

【0005】DSP55には、専用のDSPバス62を介して他のメモリ63が接続されている。また、DSP55の出力は、DAC(デジタル/アナログ変換器)56を介してアンプ、スピーカ等のサウンドシステム57に供給されている。

【0006】演奏者が、鍵盤65上で演奏操作を行なうと、演奏操作信号は1/F64を介してCPU53に伝えられる。CPU53は、メモリ61に記憶されたプログラムに従って、またメモリ61内のレジスタを用いて、指定された楽音信号を形成するために楽音パラメータを音源回路54に送る。音源回路54から発生した楽音信号は、DSP55に伝えられ、リバーブ(残響)等の効果が付与される。

【0007】DSP55は、RAM等で形成されたメモリ63を利用しつつ、所定の演奏処理を行なって効果を付与した楽音信号をDAC56に供給する。DAC56は、入力した楽音信号をアナログ信号に変換し、サウンドシステム57を介して発音させる。

【0008】なお、音色切り替えスイッチ67を操作したときは、切り替え信号が1/F64を介してCPU53に伝えられ、CPU53はメモリ61を参照して音源回路54のパラメータの変更等を行なう。

【0009】近年、半導体装置の集積度の向上にしたかい、CPUとDSPとを1チップ上に形成することが可能となってきた。CPUとDSPの1チップ化により、図7に示すような電子回路はますます普及するものと考えられる。

【0010】

【発明が解決しようとする課題】CPUとDSPとを1チップ化しても、メモリは別チップとなることが多い。CPUとそのメモリとの間には、CPUバスを設ける必要があり、DSPとそのメモリとの間にもDSPバスを設ける必要がある。したがって、CPUとDSPとの1チップ化により、半導体集積回路のピン数は大幅に増大してしまう。

【0011】ところで、DSPのメモリに対するアクセス頻度は、可能な最大アクセス頻度と比べ、かなり低い場合が多い。別の見方をすれば、DSP用メモリは遊んでいる時間が多い。しかし、DSPはDACに接続され

ているので、DACサイクルを忠実に守る必要があり、処理を待たせることはできない。

【0012】本発明の目的は、CPUとDSPを1チップ化してもピン数を大幅に増大させる必要のない楽音制御装置を提供することである。

【0013】

【課題を解決するための手段】本発明の楽音制御装置は、音源回路から供給される楽音信号を制御するための楽音制御装置であって、外部に記憶されたプログラムに従って演算処理を行なうCPUと、内部に記憶したマイクロプログラムに従って演算処理を行なうDSPと、前記CPUおよびDSPからアクセス可能なメモリと、前記CPUからアクセス可能な他の回路と、前記CPUから前記メモリへのアクセスと前記DSPから前記メモリへのアクセスとが同時に発生した時には前記CPUにウェイト信号を供給して前記DSPから前記メモリへのアクセスを優先させ、前記CPUから前記他の回路へのアクセスと前記DSPから前記メモリへのアクセスとが同時に発生した時には、前記CPUにウェイト信号を供給しないアクセス制御手段とを有する。

【0014】

【作用】CPUとDSPが同一のメモリを共用することにより、バス、ピンの数を低減し、ハードウェア資源を有効に利用することができる。

【0015】CPUおよびDSPからメモリへのアクセスは、DSPアクセスを優先することにより、DSPの処理を支障なく行なうことができる。CPUのアクセスとDSPのアクセスとが重複した場合、CPUのアクセスを待たせても、CPU処理に支障が生じることは少ない。

【0016】

【実施例】図1に、本発明の実施例による楽音制御装置を示す。CPUアドレスバス1とCPUデータバス2がCPU3に接続されている。これらのバス1、2には、インターフェイス22、24、26を介して、外部記憶装置21、パネル23、鍵盤25が接続され、CPU3との間でデータの授受を行なう。

【0017】また、バス1、2には、音源回路4が接続され、CPU3の制御を受けて楽音信号の生成を行なう。音源回路4で形成した楽音信号は、DSP5に供給される。DSP5は、供給された楽音信号に効果付与等の処理を行ない出力信号をDAC6に供給する。

【0018】DAC6は、DSP5から供給されたデジタル信号をアナログ信号に変換し、サウンドシステム7に供給して楽音を発生させる。なお、DSP5もバス1、2に接続され、CPU3の制御を受けることができる。

【0019】CPU3の制御を受ける各回路は、各回路内に複数の記憶領域を有する。CPU3と各回路とは、この記憶領域を介してデータを送受する。CPU3は、

5

CPUアドレスバス1を介して各回路にアドレスデータを出力する。このアドレスデータはmビット(m:正の整数)のデータであって、その上位nビット(n:正の整数、n < m)は各回路を指定するためのデータであり、上位(n+1)ビット目は書き込みと読み出しを区別するためのデータであり、これ以下のビットは、各回路に設けられた複数の記憶領域のいずれかを指定するためのデータである。

【0020】CPU3はアドレスデータを出力することによって、各回路の記憶領域に記憶されたデータを読み出し、この読み出したデータに基づき所定の処理を行なう。また、CPU3はアドレスデータを出力することによって、各回路を動作させるためのデータを各回路に設けられた記憶領域に書き込み、この書き込まれたデータに基づき各回路を動作させる。

【0021】RAM等で形成されたメモリ10は、アドレス端子とデータ端子とイネーブル端子とを有する。メモリ10のアドレス端子は、アドレスバスゲート14を介してCPUアドレスバス1に接続されると共に、アドレスバスゲート16を介してDSP5のDSPアドレスバス12に接続される。

【0022】メモリ10のデータ端子は、データバスゲート13を介してCPUデータバス2に接続されると共に、データバスゲート15を介してDSP5のDSPデータバス11に接続される。

【0023】メモリ10のイネーブル端子には、アドレスバスゲート14を介してデコーダ27のCPUアクセスライン18が接続されると共に、DSP5のDSPアクセスライン17が接続される。

【0024】メモリ10は、イネーブル端子にDSP5あるいはデコーダ27から信号“1”が供給されるとイネーブルされる。そして、アドレス端子に供給されるアドレスデータの上位(n+1)ビット目が読み出しを指示する場合には、アドレスデータのそれ以下のビットで指定されるアドレスに記憶されたデータを読み出してデータ端子から出力する。アドレス端子に供給されるアドレスデータの上位(n+1)ビット目が書き込みを指示する場合には、アドレスデータのそれ以下のビットで指定されるアドレスにデータ端子から入力されるデータを書き込む。

【0025】デコーダ27は、CPUアドレスバス1に接続されると共に、その出力を音源回路4、1/F2 2、24、26、DSP5の各回路に供給する。さらに、デコーダ27からCPUアクセスライン18が、バス制御回路8およびアドレスバスゲート14に接続されている。

【0026】デコーダ27は、CPU3から出力されるアドレスデータの上位nビットをデコードして、アドレスデータの上位nビットが指示する回路に信号“1”を送出し、その回路をイネーブルする。

6

【0027】デコーダ27から音源回路4、1/F2 2、24、26あるいはDSP5に信号が供給される場合、各回路は、アドレスデータの上位(n+1)ビット目が読み出しを指示する場合には、アドレスデータのそれ以下のビットで指定される領域に記憶されたデータを読み出してCPUデータバス2に出力し、アドレス端子に供給されるアドレスデータの上位(n+1)ビット目が書き込みを指示する場合には、アドレスデータのそれ以下のビットで指定される領域にCPUデータバス2から供給されるデータを書き込む。また、CPU3からメモリ10を指定するアドレスデータが出力されると、デコーダ27はCPUアクセスライン18に信号“1”を出力する。

【0028】DSP5からは、DSPアクセスライン17がデータバスゲート15、アドレスバスゲート16、メモリ10のイネーブル端子およびバス制御回路8に接続され、さらにインバータを介してデータバスゲート13、アドレスバスゲート14にも接続されている。

【0029】DSP5がメモリをアクセスするときは、DSPアクセスライン17に信号“1”を出力する。

【0030】データバスゲート13、15およびアドレスバスゲート14、16は、その端子Tに信号“1”が入力されると各ゲートに入力されるデータを通過させ、その端子Tに信号“0”が入力されると各ゲートに入力されるデータの通過を禁止する。

【0031】バス制御回路8は、CPUアクセスライン18およびDSPアクセスライン17からの信号を受け、DSPのメモリアクセスとCPUのメモリアクセスとが同時に生じたときには、CPUにウェイト信号(待機)を供給する。

【0032】クロック回路29は、システム全体を制御するクロック信号を発生し、CPU3、DSP5、バス制御回路8等にクロック信号を供給する。これによって、CPU3、DSP5等は同期して動作する。

【0033】なお、図中破線で囲んだ領域は、半導体の1チップ上に集積される機能である。ただし、図面表示の便宜のため、バスラインに関してはこの区分は厳密ではない。メモリ10がRAMの場合、CPUを動作させるプログラムはシステムの電源オンの後、外部メモリ21からメモリ10に書き込む。

【0034】CPU3は、パネル23上の操作に基づき、演奏環境を設定し、鍵盤25上の演奏操作に基づいてメモリ10に記憶されたプログラムに従って音源回路4の楽音形成パラメータ等を設定し、楽音信号を発生させる。DSP5は、音源回路4から供給される楽音信号にリバーブ(残響)等の効果を付与する。

【0035】図1に示すように、メモリ10は、DSP5およびCPU3からゲート13、14およびゲート15、16を介して共用できる構成となっている。図2は、メモリ10のメモリマップを示す。メモリ10は、

メモリアドレス\$00000から\$7ffffまでのメモリ容量を有し、\$00000から\$08000までがCPUのプログラムエリア31であり、\$08001から\$10000までがCPUのデータエリア32であり、音色データ等を記憶するワークメモリとして機能する。また、メモリアドレス\$10001から\$7ffffまではリバーブのためのメモリエリア33であり、DSP5の処理に用いられる。

【0036】DSP5またはCPU3がメモリ10をアクセスするときは、アドレス信号と同時にDSPアクセス信号またはCPUアクセス信号を発生する。メモリアクセスの重複による誤動作を防止するため、さらにバス制御回路8が設けられている。

【0037】図3は、バス制御回路の構成例を示す。クロック信号、DSPアクセス信号、CPUアクセス信号の3つの信号がAND回路36を介してJKフリップフロップ35のJ端子に接続されている。

【0038】これら3つの信号が同時に“1”となった時は、JKフリップフロップ35のJ端子に“1”が入力され、Q端子に“1”的信号が発生する。このQ端子の信号は、CPUウェイト信号であり、CPU3に供給されてCPUのメモリアクセスを待機させる。

【0039】DSPアクセス信号が“1”から“0”に変化すると、インバータ38を介してAND回路37に“1”が供給される。AND回路37の他の入力には、クロック信号が供給されているため、DSPアクセス信号が消滅した次のクロックにおいて、JKフリップフロップ35のK端子に“1”が供給される。K端子に“1”が供給されると、Q端子のCPUウェイト信号が消滅する（“0”になる）。

【0040】なお、DSPアクセス信号およびCPUアクセス信号はクロック信号と同期して発生するため、DSPアクセス信号、CPUアクセス信号が発生する時には必ずクロック信号も発生する。

【0041】このように、バス制御回路8は、DSP5のメモリアクセスとCPU3のメモリアクセスとが同時に発生した時にはCPU3のメモリアクセスを待機させ、DSP5のメモリアクセスを優先させる。

【0042】DSP5の出力は、DAC6のDACサイクルに同期しているため、DSP5の処理を遅らせるることはできない。DSP5のメモリアクセスとCPU3のメモリアクセスとが同時に発生しても、バス制御回路8によって常にDSP5の処理が優先されるため、DSP5の処理に支障は生じない。

【0043】CPU3のメモリアクセスは、DSP5のメモリアクセスと重複した時には待機させられるが、DSPのメモリアクセスは頻度が低いため、CPU3の待機時間が不当に長くなることは少ない。

【0044】図4は、DSP5の内部構成例を示す。DSP5に対する入力信号は、入力レジスタReg1に入

力され、その出力はセレクタSel11、Sel12に供給される。セレクタSel11、Sel12の出力は、乗算器Mul1に供給される。

【0045】乗算器Mul1、セレクタSel13の出力が加算器Addに供給され、その出力はレジスタReg3に供給される。レジスタReg3の出力は、出力レジスタReg4を介して出力されると共に、テンボラリレジスタReg2に供給される。テンボラリレジスタReg2の出力は、セレクタSel11、Sel12に供給される。

【0046】なお、セレクタSel12には、レジスタReg3の出力がテンボラリレジスタReg2を介さず、直接供給されている。また、セレクタSel13にもレジスタReg3の出力が供給される。セレクタSel13の他の入力には“0”が供給される。“0”が選択される時、セレクタSel13は“0”を加算器Addに供給し、加算器Addは乗算器Mul1の出力を単にレジスタReg3に伝える役割を果たす。このように、DSP5は、基本的には乗算器と加算器とがレジスタやセレクタを介して組み合わされた構成を有する。

【0047】DSP5には、係数レジスタCR、アドレスレジスタAR、マイクロプログラムレジスタMPRが設けられており、マイクロプログラムレジスタMPRのプログラムに従ってDSP5の処理を制御する。

【0048】係数レジスタCRは、乗算器Mul1での乗算に必要な乗算係数を供給する。アドレスレジスタARはアドレスコントロールACを介して相対アドレスを物理アドレスに変換する。マイクロプログラムレジスタMPRから読み出／書き信号が発生した時は、タイミングコントロールTCLを介してDSP5アクセス信号が発生する。

【0049】また、アドレスコントロールACからの物理アドレスは、タイミングコントロールTCLを介してDSP5アドレス信号を形成する。また、レジスタReg3の出力もタイミングコントロールTCLを介してDSPデータとして出力する。

【0050】DSP5には、CPU3からCPUアドレスバス1およびCPUデータバス2を介してデータおよびアドレスが供給され、係数レジスタCR、アドレスレジスタAR、マイクロプログラムレジスタMPRに供給される。また、クロック信号もクロック回路29から供給される。

【0051】DSP5は、図中左側に示した部分によって同一の演算処理を繰り返し行なう。その際、メモリに対するアドレスを変化させるため、アドレスコントロールACは1回の処理毎にアドレスを1デクリメントする。アドレスが最小値に達した時は、最大値にシャンフルする。

【0052】図5は、DSP5内の係数レジスタCR、アドレスレジスタAR、マイクロプログラムレジスタM

PRの構成例を示す。マイクロプログラムレジスタMRは128ステップを有するものとする。

【0053】クロック信号に応じてマイクロプログラムレジスタに記憶されたマイクロプログラムが“0”から“127”に向かって順次進行し、“127”に達した後は、再び“0”に戻る。

【0054】アドレスレジスタARは、マイクロプログラムレジスタに同期して動作し、たとえばステップ1のマイクロプログラム「書き込み」に対応してメモリアドレス\$10000が記憶されている。

【0055】また、ステップ3の読み出のマイクロプログラムに応じてメモリアドレス\$3ffffが記録されている。すなわち、メモリのアドレス\$3ffffの情報が読み出され、テンポラリレジスタTemp1に入力される。

【0056】同様、マイクロプログラムのステップ7においては、メモリのアドレス\$50000からデータが読み出され、テンポラリレジスタTemp2に入力される。ステップ8においては、メモリのアドレス\$7ffffからデータが読み出され、テンポラリレジスタTemp3に入力される。

【0057】このように、マイクロプログラムの進行に従って、アドレスレジスタが指定するメモリアドレスに書き込み、読み出が行なわれる。なお、係数レジスタCRもマイクロプログラムに同期して変化する。

【0058】なお、マイクロプログラムが一巡した時は、メモリアドレスを変化させるために、アドレスレジスタARの出力するアドレスはアドレスコントロールACで1デクリメントする。

【0059】DSP5でリバーブの効果を付与する場合は、音源回路4が発生した楽音信号が図2に示すメモリ10のリバーブのためのメモリエリア33に書き込まれ、一定の遅延時間を経過した後、DSPのマイクロプログラムに従って読み出され、図4の示すようなDSP演算処理回路によってリバーブの効果を与えられ、DAC6に出力される。

【0060】DSPにおけるセレクタやラッチの選択等は、予めマイクロプログラムに設定されているため、自動的に行なわれ、その度にメモリを参照する必要はない。したがって、DSPがメモリをアクセスする頻度はクロックと比べ、大幅に低いものとなる。CPU3からメモリ10へのアクセスは、DSP5がメモリ10をアクセスしていない間に行なわれる。

【0061】なお、DSPの構成および動作の詳細は、本出願人が先に提出した特願平5-57504号に示されている。次に、上述した実施例の動作を説明する。

【0062】図1において、CPU3がメモリ10をアクセスする場合には、CPU3はメモリ10を指定するアドレスデータを出力する。すると、デコーダ27は、このアドレス信号をデコードしてCPUアクセスライン

18に信号“1”を出力する。この信号はアドレスバスゲート14を介してメモリ10のイネーブル端子に供給され、メモリ10はイネーブルされる。

【0063】一方、DSP5がメモリ10をアクセスする場合には、DSP5はDSPアクセスライン17に信号“1”を出力する。この信号はメモリ10のイネーブル端子に供給され、メモリ10はイネーブルされる。

【0064】ここで、CPU3のメモリアクセスとDSP5のメモリアクセスとが同時に発生した場合には、データバスゲート15、アドレスバスゲート16の端子T

10にはDSPアクセスライン17に出力される信号“1”が入力されるので、DSPデータバス11とDSPアドレスバス12はメモリ10に接続される。

【0065】一方、データバスゲート13、アドレスバスゲート14の端子TにはDSPアクセスライン17に出力される信号“1”が反転されて入力されるので、CPUアドレスバス1とCPUデータバス2はメモリ10に接続されない。このとき、バス制御回路は、CPU3のメモリアクセスとDSP5のメモリアクセスとが同時に発生したことを検知し、DSPアクセスラインに信号

20 “1”が出力されている間、CPU3にウェイト信号waitを出力する。

【0066】CPU3は、バス制御回路8からウェイト信号waitが入力されている間、メモリアクセスの状態を保持する。DSP5のメモリアクセスが終了すると、データバスゲート13、アドレスバスゲート14の端子TにはDSPアクセスライン17に出力される信号“0”が反転されて入力されるので、CPUアドレスバス1とCPUデータバス2はメモリ10に接続され、CPUのメモリアクセスが行なわれる。

【0067】図6は、DSPとCPUのメモリアクセスに関するタイミングチャートを示す。図中、最上段にDACサイクルを示す。このDACサイクル内に2段目のマイクロプログラムが実行される。DACサイクルは、128ステップに対応するとする。

【0068】図中、3段目に示すクロック信号は、マイクロプログラムの各ステップ毎に1サイクルの変化を示す。DSPからメモリへのアクセスは、4段目のアクセス信号を伴って5段目のアドレス信号を発生することによって行なわれる。

【0069】図6においては、マイクロプログラムの第1ステップ、第3ステップ、第7ステップ、第8ステップにおいて、DSPからメモリへのアクセスが行なわれている。これらのステップの後半において、5段目のDSPデータ（メモリ10に書き込まれるデータあるいはメモリ10から読み出されたデータ）が発生する。

【0070】CPU3からメモリ10へのアクセスも、7段目のCPUアクセス信号を伴って8段目のCPUアドレス信号が発生することにより行なわれる。図示の場合、マイクロプログラムの第1ステップにおいて、DS

PアクセスとともにCPUアクセスが生じている。このため、バス制御回路8が最下段のCPUウェイト信号を発生する。

【0071】CPUウェイト信号は、第2ステップにおいて、DSPアクセスが消滅することによって消滅する。したがって、第2ステップにおいて、CPUアクセスが行なわれ、その後半で9段目のCPUデータが発生する。

【0072】第4ステップにおいてCPUアクセスが発生しているが、この場合にはDSPアクセスが発生していないため、CPUからメモリへのアクセスはそのまま行なわれる。

【0073】第7ステップにおいて、CPUアクセスがDSPアクセスと重複して発生している。この場合、CPUウェイト信号の発生によって第8ステップに移動するが、第8ステップにおいてもDSPアクセスが発生している。したがって、第8ステップにおいてもCPUウェイト信号が発生する。

【0074】第9ステップになると、DSPアクセスが消滅するため、CPUウェイト信号も消滅し、CPUからメモリへのアクセスが行なわれる。このようなタイミング制御により、同一のメモリをDSPとCPUによって共用することができる。DSPのメモリアクセスは常に優先されるため、DSPの処理に支障が生じることはない。CPUからのメモリアクセスは、DSPのメモリアクセスと重複したときは待機させられるが、DSPのメモリアクセスが消滅すると直ちに実行される。

【0075】図1に示すように、破線内を1チップ化した時、この半導体集積回路からメモリ10に対するピンはアドレス用とデータ用の一組のみでよく、DSP専用メモリとCPU専用メモリを用いた場合と比べ、ピン数は大幅に減少する。

【0076】なお、楽音制御装置としてDSPとCPUを1個ずつ用いる場合を説明したが、複数個のDSPと複数個のCPUを用いてもよい。DSPによってリバーブ効果を付与する場合を説明したが、DSPの演算はリバーブに限らず、どのようなものであってもよい。

【0077】以上実施例に沿って本発明を説明したが、本発明はこれらに制限されるものではない。たとえば、種々の変更、改良、組み合わせ等が可能なことは当業者に自明であろう。

【0078】

【発明の効果】以上説明したように、本発明によれば、DSPとCPUが同一のメモリを共用できるため、ハードウェア資源の利用効率を向上することができる。また、DSPのメモリアクセスをCPUのメモリアクセスよりも優先させることにより、DSPの処理を支障なく行うことができる。

【0079】また、同一のメモリをDSPとCPUで共用することにより、回路構成を簡単化することができる。

る。DSPとCPUを1チップ上に集積化した場合、この集積回路装置のピン数を低減することができる。

【図面の簡単な説明】

【図1】本発明の実施例による楽音制御装置を示すブロック図である。

【図2】図1の実施例におけるメモリのメモリマップである。

【図3】図1の実施例におけるバス制御回路の構成例を示すブロック図である。

【図4】図1の実施例に用いるDSPの構成例を示すブロック図である。

【図5】図4のDSPにおける係数レジスタ、アドレスレジスタ、マイクロプログラムレジスタの構成例を示す概略図である。

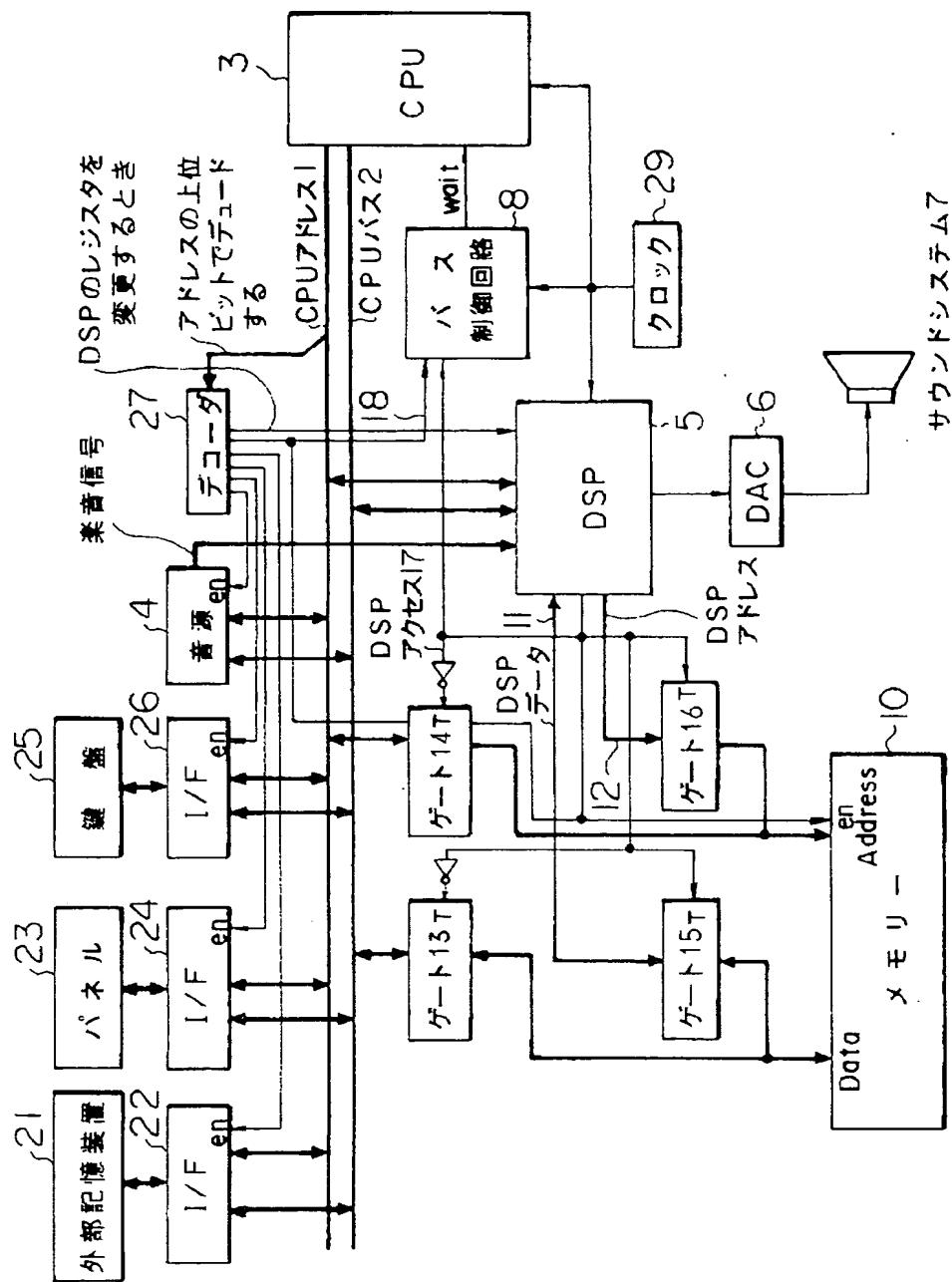
【図6】図1の実施例の動作を説明するためのタイミングチャートである。

【図7】従来の技術による楽音制御装置の構成例を示すブロック図である。

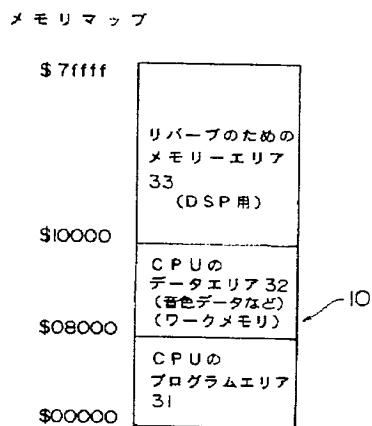
【符号の説明】

20	1 CPUアドレスバス
	2 CPUデータバス
	3 CPU
	4 音源回路
	5 DSP
	6 DAC
	7 サウンドシステム
	8 バス制御回路
	10 メモリ
	13、15 データバスゲート
30	14、16 アドレスバスゲート
	17 DPSアクセスライン
	18 CPUアクセスライン
	21 外部記憶装置
	22、24、26 I/F
	23 パネル
	25 鍵盤
	27 デコーダ
	29 クロック発生回路
	35 JKフリップフロップ
	36、37 AND回路
	38 インバータ
	CR 係数レジスタ
	AR アドレスレジスタ
	MPR マイクロプログラムレジスタ
	TCL タイミングコントロール
	Reg レジスタ
	Sel セレクタ
	Mul 乗算器
	Ad 加算器

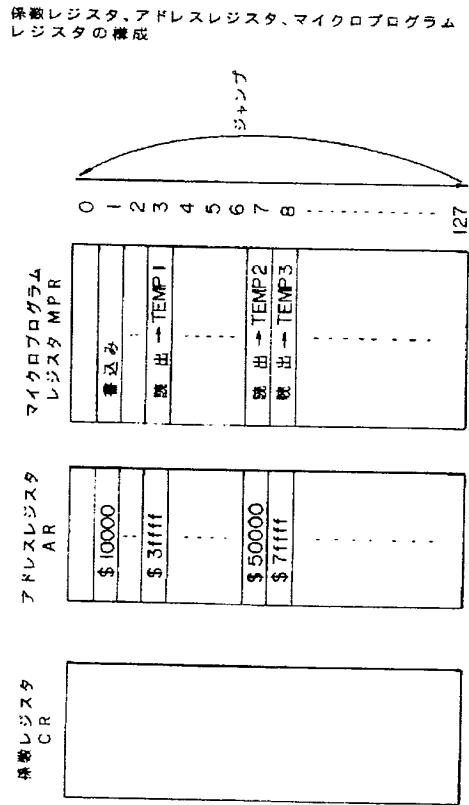
〔図1〕



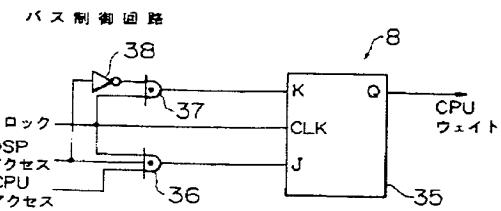
【図2】



【図5】

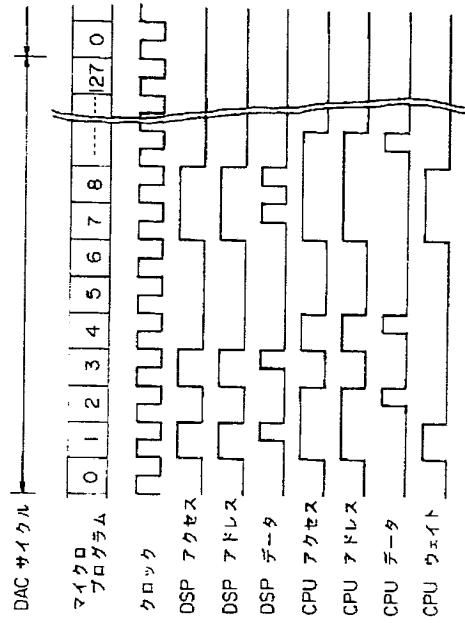


【図3】

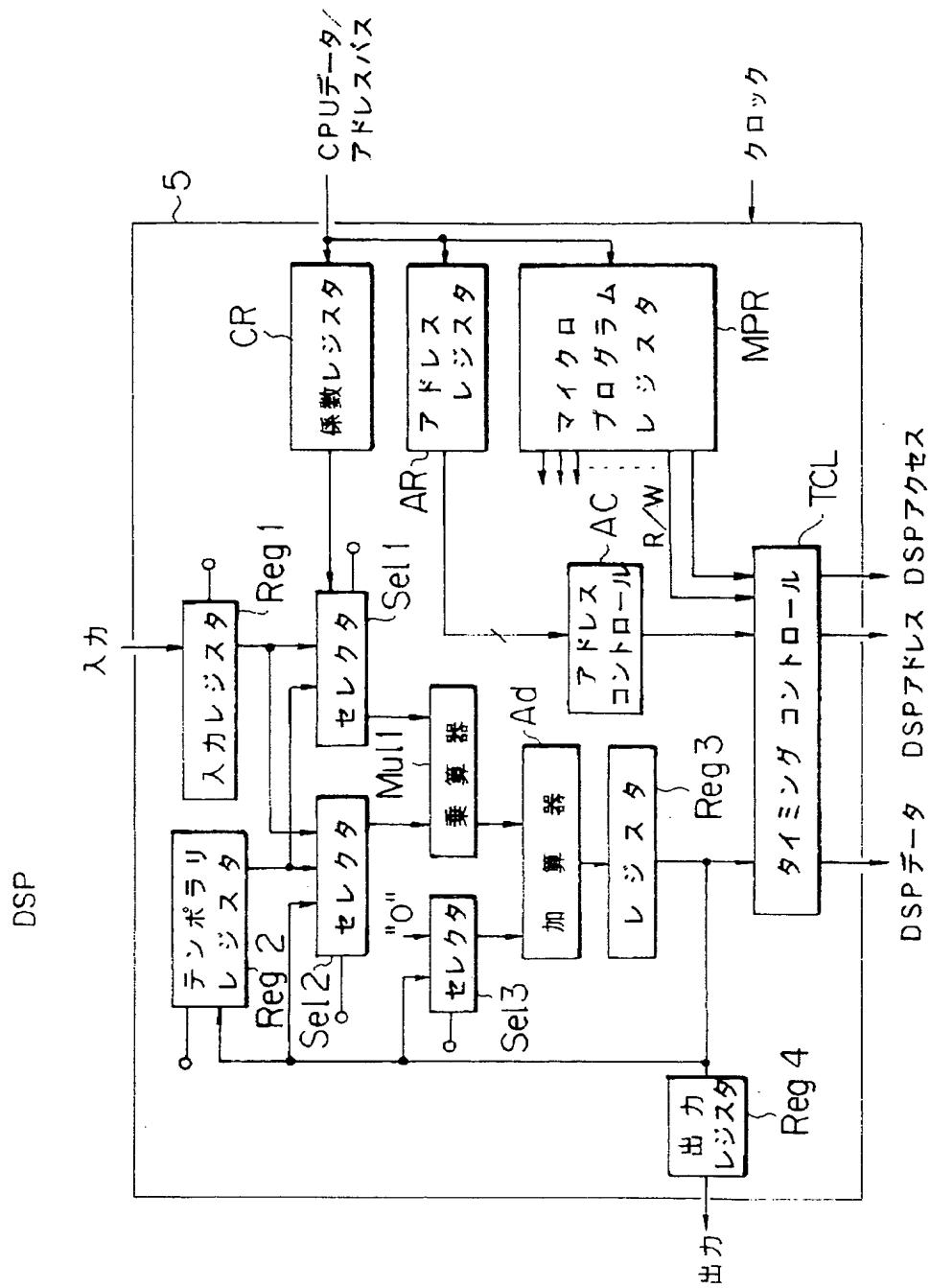


【図6】

タイミングチャート



[图4]



【図7】

従来技術

